PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:

McLampy *et al.*

Serial No.: 09/941,229

Filed: August 28, 2001

Confirmation No.: 5275

Group Art Unit: 2131

Examiner: Sherkat, Arezoo

TKHR Ref: 050115-1050

For: **SYSTEM AND METHOD FOR PROVIDING ENCRYPTION FOR REROUTING OF REAL TIME MULTI-MEDIA FLOWS**

APPEAL BRIEF UNDER 37 C.F.R. §1.192

Mail Stop Appeal Brief - Patents
Commissioner of Patents and Trademarks
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

This is an appeal from the decision of Examiner Arezoo Sherkat, Group Art Unit 2131 mailed January 17, 2007, rejecting claims 45-62, 67, and 70-73 in the present application and making the rejection FINAL.

I. REAL PARTY IN INTEREST

The real party in interest is Primary Networks, Inc. d/b/a Acme Packet, Inc. and having a principal place of business at 130 New Boston Street, Woburn, Massachusetts 01801, U.S.A.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF THE CLAIMS

Claim 45-62, 67, and 70-73 are pending in this application. Through prosecution of this matter, claims 1-44, 63-66 and 68-69 have been canceled without prejudice, waiver, or disclaimer. Claims 45-62, 67, and 70-73 were rejected by the FINAL Office Action and are the subject of this appeal. A FINAL Office Action, dated January 17, 2007, affirmed the rejection.

IV. STATUS OF AMENDMENTS

All amendments submitted before the mailing date of the FINAL Office Action have been entered. No amendments have been submitted since the mailing date of the FINAL Office Action. A copy of the current claims is attached hereto in the Appendix.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Embodiments of the claimed subject matter are illustrated in FIGS. 1-4 and are discussed in the specification at least at pages 5-21. Generally speaking, Applicants claim systems and methods for encrypting multi-media data flow packets. The encryption system of the present invention can be implemented in software, firmware, hardware, or a combination thereof. In the preferred embodiment of the invention, which is intended to be a non-limiting example, a portion of the encryption system is implemented in software that is executed by a computer.

The claimed inventions are summarized below with reference numerals and references to the written description ("specification") and drawings. The subject matter described in the following appears in the original disclosure at least where indicated, and may further appear in other places within the original disclosure.

Embodiments of the invention, such as those defined by independent claim 45, define a method of encrypting multi-media data flow packets. (*See, e.g.,* FIG. 4.) The

method of claim 45 comprises receiving a series of multi-media data flow packets, each packet comprising a sequence number. (*See, e.g., specification,* page 14, lines 5-7.) The method of claim 45 further comprises storing the series of multi-media data flow packets in a jitter buffer. (*See, e.g., specification,* page 17, lines 11-20.) The method of claim 45 further comprises re-sequencing the series of multi-media data flow packets into a pseudo-random order. (*See, e.g., specification,* page 14, lines 8-13.) The method of claim 45 further comprises transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order. (*See, e.g., specification,* page 17, line 21 to page 18, line 5.)

Independent claim 52 describes a computer readable medium for encrypting multi-media data flow packet. (*See, e.g., specification,* page 7, line 12 to page 8, line 6.) The computer readable medium of claim 52 comprises a program for receiving a series of multi-media data flow packets. (*See, e.g., specification,* page 14, lines 5-7.) The computer readable medium of claim 52 further comprises a program for storing the series of multi-media data flow packets in a jitter buffer. (*See, e.g., specification,* page 17, lines 11-20.) The computer readable medium of claim 52 further comprises a program for re-sequencing the series of multi-media data flow packets into a pseudo-random order. (*See, e.g., specification,* page 14, lines 8-13.) The computer readable medium of claim 52 further comprises a program for transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order. (*See, e.g., specification,* page 17, line 21 to page 18, line 5.)

Independent claim 59 describes a system for encrypting multi-media data flow packets. (*See, e.g., specification,* page 14, lines 5-7.) The system of claim 59 comprises a transceiver. (*See, e.g.,* FIGS. 1, 3.) The system of claim 59 further comprises software stored within said first endpoint defining functions to be performed

by the system. (*See, e.g., specification,* page 7, lines 6-19.) The system of claim 59 further comprises a processor (*see, e.g., specification,* page 7, lines 12-17; FIG. 3.) configured by said software to perform the steps of receiving a series of multi-media data flow packets (*see, e.g., specification,* page 17, lines 11-20), storing the series of multi-media data flow packets in a jitter buffer (*see, e.g., specification,* page 17, lines 11-20), re-sequencing the series of multi-media data flow packets into a pseudo-random order (*see, e.g., specification,* page 14, lines 8-13), and transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order (*see, e.g., specification,* page 17, line 21 to page 18, line 5).

Independent claim 67 describes a method of encrypting a series of multi-media data flow packets. (*See, e.g., specification,* FIG. 4.) The method of claim 67 comprises receiving a series of multi-media data flow packets belonging to a first flow, each packet in the series having the same port address. (*See, e.g., specification,* page 15, lines 17 to page 17, line 9.) The method of claim 67 further comprises generating a pseudo-random sequence of numbers, the sequence associated with the port address. (*See, e.g., specification,* page 14, lines 10-13; page 17, lines 3-5.) The method of claim 67 further comprises replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence. (*See, e.g., specification,* page 17, lines 3-9.) The method of claim 67 further comprises transmitting each packet to a receiver. (*See, e.g., specification,* FIG. 4 and related text.)

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The FINAL Office action rejected claims 45-62 under 35 U.S.C. § 102(e) as allegedly being anticipated by *Fink* et al. (U.S. Pat. No 6,826,684, hereinafter "*Fink*"). Claims 67 and 70-73 were rejected under 35 U.S.C. §103(a) as allegedly being

unpatentable over *Fink*, in view of *Akiyama et al.* (U.S. Pat. No. 5,623,548, hereinafter "*Akiyama*").

VII. ARGUMENT

For at least the reasons set forth herein, Applicants disagree and request that the rejections be overturned.

A. Claim Rejections Under 35 U.S.C. § 102

Turning now to the substantive rejections, claims 45-62 stand rejected under 35 U.S.C. § 102(e) as allegedly being anticipated by *Fink.* Applicants traverse the rejection.

The Fink Reference

Fink describes a bastion host for a local area network (LAN). The bastion host processes packets to be transferred from the LAN to a wide area network (WAN). The bastion host intercepts packets originating from a host on the LAN, the packets being destined for transmission over the WAN, extracts bits from predetermined fields from each packet header to form one or more blocks for translation, masks bits from the one or more blocks that vary rapidly packet to packet, applies a predetermined encryption algorithm to translate the one or more blocks after masking, and reinserts bits from the translated block back into the packet header. (*See Fink*, Abstract.) *Fink*, however, does not disclose re-sequencing a series of multi-media data flow packets into a pseudo-random order and transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order. *Fink* also fails to disclose replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence.

Independent Claims 45, 52, and 59 Are Patentable Over Fink

A proper rejection of a claim under 35 U.S.C. §102 requires that a single prior art reference disclose each element of the claim. *See, e.g., W.L. Gore & Assoc., Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 U.S.P.Q. 303, 313 (Fed. Cir. 1983).

Applicants respectfully submit that *Fink* fails to teach, disclose or suggest at least the feature of "***re-sequencing the series of multi-media data flow packets into a pseudo-random order; and transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order***" as recited in claims 45, 52, and 59. The *Fink* reference discloses a system which receives packets, encrypts portions of an IP packet header, including "packet sequencing information" and re-transmits the encrypted packet. The Office Action alleges that encryption corresponds to "re-sequencing the series of multi-media data flow packets into a pseudo-random order" as recited in claims 45, 52, and 59. (*See* FINAL Office Action mailed 1/17/07, p. 4.) Applicants respectfully disagree.

First, Applicants disagree that "encryption" corresponds to "re-sequencing". *Fink* appears to disclose conventional block cipher techniques by which an input byte array is transformed according to a key rather than any sort of encryption that involves re-sequencing. Next, even assuming, *arguendo*, that *Fink* does disclose re-sequencing bytes, claims 45, 52, and 59 do not recite re-sequencing bytes within a packet, but instead recite re-sequencing the packets in a series of received packets.

Applicants have examined *Fink* and find no discussion of sequencing or re-sequencing. In the FINAL Office Action (mailed 1/17/07), the Examiner cites the following text from the *Fink* reference:

> Table 2 illustrates how the contents of the encrypted byte array 310 are **repacked** into the original TCP/IP packet header, thereby replacing the old (original) information. The packet header at this point **is said to be translated**.

(Col. 8, lines 39-44) Applicants respectfully disagree that "translated," as recited in the text above, equates to "re-sequenced," as alleged by the FINAL Office Action. Applicants refer to the following text in the *Fink* reference:

> [A]ll packets matching a given destination address are consistently **translated, or mapped**, to a fixed "other" destination address for a given interval of time.

(Col. 3, lines 2-5) As supported by the text above, the term translated refers to mapping one value to another value. The term does not appear to relate to re-sequencing, as alleged on page 3 of the FINAL Office Action.

Fink also does not disclose, teach, or suggest the feature, "***transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order***." The only description in *Fink* of the transmission of the encrypted packets is "[o]nce translated, this encrypted packet is transmitted across the Internet 36." (*Fink*, Col. 7, lines 15-20.) The discussion of the receiver fails to say anything about the sequence in which packets are transmitted: "if the sending enclave is recognized as a trusted enclave, the receiving ASD peer 35 restores the packet in accordance with the prearranged protocol. The result of this process is a restored packet identical to the original packet created by the sending host 31." (*Fink*, Col. 7, lines 20-25.)

The Office Action indicates that "such restoration is required because packet header information such as sequence number has been randomized/encrypted before transmission." Applicants respectfully submit that this "restoration" refers only to decryption, and does not relate at all to the packet's position in a received sequence. Applicants do not disagree that the receiver must restore the packet after transmission, and also agree that the transmitter changes the contents of the packets by encryption. However, neither of these features in the *Fink* reference corresponds to the features of "re-sequencing the series of multi-media data flow packets" and "transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order" as

recited in claims 45, 52, and 59. Accordingly, *Fink* fails to teach of "re-sequencing" and "transmitting...in the re-sequenced order."

For at least the reason that *Fink* fails to disclose, teach or suggest the above-recited features, Applicants respectfully submit that *Fink* does not anticipate claims 45, 52, and 59. Therefore, Applicants request that the rejection of claims 45, 52, and 59 be withdrawn.

Dependent Claims 46-51, 53-58, and 60-62 are Patentable

Since independent claims 45, 52, and 59 are allowable, Applicants respectfully submit that claims 46-51, 53-58, and 60-62 are allowable for at least the reason that each depends from an allowable claim. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q. 2d 1596, 1598 (Fed. Cir. 1988). Therefore, Applicants respectfully request that the rejection of claims 46-51, 53-58, and 60-62 be withdrawn.

B. Claim Rejections Under 35 U.S.C. § 103

Claims 70-73 stand rejected under §103(a) as allegedly being unpatentable over *Fink* in view of *Akiyama*. Applicants traverse the rejection.

The Fink Reference

The *Fink* reference was discussed in the preceding section.

The Akiyama Reference

The *Akiyama* reference describes a transformation pattern generating device and encryption function device. *Akiyama* seeks to provide to provide a transformation pattern generating device and encryption function device which can disable differential attack through the key-dependent alteration of a permutation mechanism or substitution

table structure in the encryption function and realize a cryptosystem virtually unbreakable through the differential attack. (*See Akiyama*, Summary of the Invention.) *Akiyama*, however, fails to disclose replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence.

Independent Claim 67 is Patentable over Fink, in View of Akiyama

It is well established at law that, for a proper rejection of a claim under 35 U.S.C. §103 as being obvious based upon a combination of references, the cited combination of references must disclose, teach, or suggest, either implicitly, all elements/features/steps of the claim at issue. *See, e.g., In re Dow Chemical*, 5 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1988); In re Keller, 208 U.S.P.Q.2d 871, 881 (C.C.P.A. 1981). As can be verified by a review of the *Fink* and *Akiyama* references, neither reference discloses, teaches, or suggests generating a product from the corresponding number in the sequence and the size of the sequence. As such, Applicants submit that *Fink* nor *Akiyama*, alone or in combination, discloses, teaches, or suggests the feature "***replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence***," as recited in claim 67.

Accordingly, as the proposed combination does not teach at least the above-described feature recited in claim 67, a prima facie case establishing an obviousness rejection has not been made. Thus, claim 67 is not obvious under the proposed combination of *Fink* in view of *Akiyama*, and the rejection should be withdrawn.

Dependent Claims 70-73 are Patentable

Since claim 67 is allowable, Applicants respectfully submit that claims 70-73 are allowable for at least the reason that each depends from an allowable claim. *In re Fine*,

837 F.2d 1071, 5 U.S.P.Q. 2d 1596, 1598 (Fed. Cir. 1988). Therefore, Applicants respectfully request that the rejection of claims 70-73 be withdrawn.

C. Additional Remarks

Applicants would like to note that during the course of prosecution of the present application, the Examiner has completely failed to address various amendments made to claim 67 (even while accepting the amendments) and maintains an improper rejection in the FINAL Office Action. (*See* FINAL Office Action mailed 1/17/07, p. 7.) Applicants, in fact, amended claim 67 in the response to the Office Action mailed 1/11/06 to recite the following: "replacing the port address in each packet with the corresponding number in the sequence or the product of the corresponding number in the sequence and the size of the sequence." In the Office Action mailed 6/23/06, the Examiner entered the amendments (see Response to Amendment section in Office Action mailed 6/23/06) but failed to provide a proper rejection as the rejection was based on the claim language prior to the amendment. (See Office Action mailed 6/23/06, p. 8). To further prosecution, Applicants further amended claim 67 in response to the non-final Office Action mailed 6/23/06 to recite: "replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence." The FINAL Office Action, however, maintains the §103 rejection based on the following language: "replacing the port address in each packet with the corresponding number in the sequence."

VIII. CONCLUSION

In summary, it is Applicants' position that Applicants' claims are patentable over the cited references and that the rejection of these claims should be withdrawn. Applicants therefore respectfully requests that the Board of Appeals overturn the Examiner's rejection and allow Applicants' pending claims.

Respectfully submitted,

/Karen G. Hazzah/

Karen G. Hazzah, Reg. No. 48,472

THOMAS, KAYDEN, HORSTEMEYER & RISLEY, L.L.P.
100 Galleria Parkway NW
Suite 1750
Atlanta, Georgia 30339
(770) 933-9500

CLAIMS APPENDIX

1-44. (Cancelled)

45. (Previously Presented) A method of encrypting multi-media data flow packets, comprising the steps of:

receiving a series of multi-media data flow packets, each packet comprising a sequence number;

storing the series of multi-media data flow packets in a jitter buffer;

re-sequencing the series of multi-media data flow packets into a pseudo-random order; and

transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order.

46. (Previously Presented) The method of claim 45, wherein said re-sequencing uses a randomization code that is algorithmically predictable if a key to said randomization code is known.

47. (Previously Presented) The method of claim 45, further comprising the step of performing bit manipulation within said first multi-media data flow packet.

48. (Previously Presented) The method of claim 47, wherein said step of performing bit manipulation is performed by using a bit-size operation that is restorable.

49. (Previously Presented) The method of claim 48, wherein said bit-size operation comprises negation.

50. (Previously Presented) The method of claim 45, further comprising the step of pseudo-randomly shuffling a destination address of each of the multi-media data flow packets.

51. (Previously Presented) The method of claim 50, wherein said destination address is a destination port address of said second endpoint.

52. (Previously Presented) A computer readable medium for encrypting multi-media data flow packets, the program for performing the steps of:

receiving a series of multi-media data flow packets;

storing the series of multi-media data flow packets in a jitter buffer;

re-sequencing the series of multi-media data flow packets into a pseudo-random order; and

transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order.

53. (Previously Presented) The computer readable medium of claim 52, wherein said re-sequencing uses a randomization code that is algorithmically predictable if a key to said randomization code is known.

54. (Previously Presented) The computer readable medium of claim 52, the program further comprising logic for performing the step of performing bit manipulation within said first multi-media data flow packet.

55. (Previously Presented) The computer readable medium of claim 54, wherein said step of performing bit manipulation is performed by using a bit-size operation that is restorable.

56. (Previously Presented) The computer readable medium of claim 55, wherein said bit-size operation comprises negation.

57. (Previously Presented) The computer readable medium of claim 52, the program further comprising logic for performing the step of pseudo-randomly shuffling a destination address of each of the multi-media data flow packets.

58. (Previously Presented) The computer readable medium of claim 57, wherein said destination address is a destination port address of said second endpoint.

59. (Previously Presented) A system for encrypting multi-media data flow packets, comprising:

a transceiver;

software stored within said first endpoint defining functions to be performed by the system; and

a processor configured by said software to perform the steps of:

receiving a series of multi-media data flow packets;

storing the series of multi-media data flow packets in a jitter buffer;

re-sequencing the series of multi-media data flow packets into a pseudo-random order; and

transmitting each multi-media data flow packet in the re-sequenced series in the re-sequenced order.

60. (Previously Presented) The system of claim 59, wherein said re-sequencing uses a randomization code that is algorithmically predictable if a key to said randomization code is known.

61. (Previously Presented) The system of claim 59, processor configured by said software to perform the step of pseudo-randomly shuffling a destination address of each of the multi-media data flow packets.

62. (Previously Presented) The system of claim 61, wherein said destination address is a destination port address of said second endpoint.

63-66. (Cancelled)67. (Previously Presented) A method of encrypting a series of multi-media data flow packets, comprising the steps of:

receiving a series of multi-media data flow packets belonging to a first flow, each packet in the series having the same port address;

generating a pseudo-random sequence of numbers, the sequence associated with the port address;

replacing the port address in each packet with the product of the corresponding number in the sequence and the size of the sequence; and

transmitting each packet to a receiver.

68. (Cancelled)

69. (Cancelled)

70. (Previously Presented) The method of claim 67, wherein the generating step uses a randomization code that is predictable if a key to the randomization code is known.

71. (Previously Presented) The method of claim 70, wherein the key is known to the receiver.

72. (Previously Presented) The method of claim 67, wherein the size of the sequence is known to the receiver.

73. (Previously Presented) The method of claim 67, wherein the port address comprises a destination port address.

EVIDENCE APPENDIX

(None)

RELATED PROCEEDINGS APPENDIX

(None)